

AA 2019-20

FONDAMENTI DI INFORMATICA

LABORATORIO
IV

[prof.ssa RAFFAELA MIRANDOLA]



POLITECNICO
MILANO 1863

Esercizio di riscaldamento (I)

Scrivere un programma che implementi la funzione `numeroPrimo(int n)`. La funzione deve restituire 0 se n non è primo, 1 se è primo.

Esercizio di riscaldamento (II)

A partire dalla funzione scritta in precedenza, la si modifichi in modo che calcoli quanti numeri primi ci sono in un intervallo $[n ; m]$.

Esercizio 1

Si scriva una funzione che restituisca 1 se un numero complesso è situato nell'origine, altrimenti 0. La funzione segua il prototipo:

```
int isOrigin(struct Complex n) ;
```

Suggerimenti:

- definire opportunamente lo struct **Complex**.

Esercizio 2

Si scriva una funzione che legga da tastiera una stringa fino alla pressione del tasto invio e restituisca il numero di caratteri letti. La stringa da caricare è passata *per indirizzo*.

```
int leggiStringa(char *s) ;
```

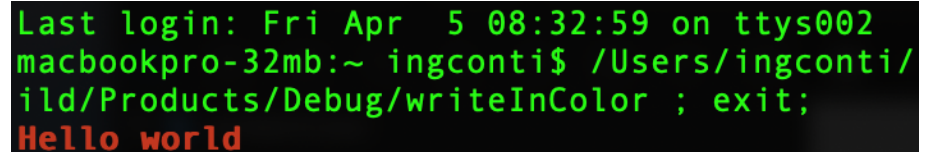
Suggerimenti:

- si utilizzi `getchar()`

Esercizio 3 – introduzione

Nella shell di Unix è possibile scrivere a colori. Di seguito è un breve esempio di come, sfruttando le funzioni, sia possibile stampare a video una stringa del colore desiderato.

```
int main ()
{
    printf("\033[1;31m");
    printf("Hello world\n");
    printf("\033[0m;");
    return 0;
}
```

A terminal window with a black background. The text is displayed in green, except for the output 'Hello world' which is red. The text shows the login session, the command to run the program, and the resulting output.

```
Last login: Fri Apr  5 08:32:59 on ttys002
macbookpro-32mb:~ ingconti$ ./Users/ingconti/
ild/Products/Debug/writeInColor ; exit;
Hello world
```

Esercizio 3

Sfruttando quanto scritto nell'esercizio precedente, si scriva una funzione che legga da tastiera una stringa di testo nei formati "red", "green" o "blue" e stampi il vostro nome nel colore indicato. Quest'effetto deve poter essere ottenuto sia nel caso il colore passato in input sia scritto maiuscolo che nel caso sia scritto in minuscolo o in qualsiasi combinazione di maiuscole e minuscole. ("normalizzazione", es.: `red`, `RED`, `Red`, `rED` → `red`)

Suggerimenti:

- si utilizzi il confronto fra stringhe
- Per una lista di colori si consulti: <https://bluesock.org/~willkg/dev/ansi.html>
- si definisca una funzione `char* toLowerCase(char *s)` che renda la normalizzazione dell'input un modulo a sé stante

Esercizio 4 – introduzione

Ci si immagini di dover implementare una struttura dati per una rubrica. Ogni contatto presente in rubrica avrebbe la forma di uno struct inclusivo delle informazioni anagrafiche:

- Cognome
- Nome
- Data di nascita
- Età
- Numero di telefono

...dove Età è frutto di un calcolo rispetto alla data di nascita e alla data attuale (si supponga *hardcoded*). La data di nascita, inoltre, potrebbe essere esplicitata dall'utente in formato italiano (dd/mm/aaaa) o in formato americano (aaaa/mm/gg), e questo imporrebbe un controllo software sul dato.

Esercizio 4

Tenendo conto di quanto all'introduzione, si scriva un programma che implementi una rubrica del telefono minimale. La rubrica deve fornire le seguenti funzionalità:

- inserire/cancellare contatti
- stampare elenco contatti con informazioni
- ricercare un contatto
- effettuare statistiche riguardanti l'età (media, mediana) e nomi dei contatti (numero di occorrenze)

Suggerimenti:

- si supponga nota la lista dei contatti.
- si riutilizzino eventuali funzioni già implementate x manipolazione stringhe e/o si scrivano ex novo (**vietato usare <stdlib.h>**)
- s'imposti un numero massimo CONT_MAX di contatti